In The Specification

Applicant is amending the specification in this application under 37 CFR 1.111 and 37 CFR 1.121. As required by the Examiner, Applicant has added application serial numbers to the two applications incorporated by reference on page 15 of the specification as originally filed to sufficiently identify the referenced material. In addition, the legend "Prior Art" has been added to Table 1 on page 5 of the specification as originally filed, along with a few ministerial changes. Applicant is not adding new matter with these amendments.

Please amend the specification with the selected replacement paragraphs on the following pages.

Included with this amendment is a version of these replacement paragraphs with markings that show the changes made to the original paragraphs in the application.

Paragraph at p. 2, l. 9-12.

N-nary logic is a new dynamic logic design style fully described in a copending patent application, U.S. Pat. App. Ser. No. 09/019355, filed 2-5-98, now U.S. Pat. No. 6,066,965, and titled "Method and Apparatus for a N-Nary logic Circuit Using 1-of-4 Signals", which is incorporated herein for all purposes and is hereinafter referred to as "The N-nary Patent."

Paragraph at p. 5, l. 3-13.

Table 1 provides a summary of the logic states supported by existing simulation tools currently used to evaluate and verify typical binary-based logic designs.

Table 1-Prior Art

Model state level	Logic states supported	Definitions
2-state	0, 1	
3-state	0, 1, X	X=uninitialized
4-state	0, 1, X, Z	Z="undriven" or high-impedance
9-state	0S, 1S, XS, 0R, 1R, XR, 0Z, 1Z, XZ	S="Strongly" driven; R="weakly" driven; Z=not driven, value of wire=last known value
MVL-9	0S, 1S, XS, 0R, 1R, XR, Z, UI, UD	Z=not driven; last value of wire ignored UI=uninitialized; last value of wire ignored UD=undefined; last value of wire ignored
12-state	0S, 1S, XS, 0R, 1R, XR, 0Z, 1Z, XZ, 0I, 1I, XI	I=Indeterminate

Paragraph at p. 15, l. 8-23.

The simulation environment files 130 can be written in any programming language, but in a preferred embodiment, are written in a version of ANSI C++ that is compatible with the g++ compiler 124 within the Design Tool Compiler 120. The simulation environment files provide the interface between the operator and the simulated logic, and enable the operator to specify test-related variables such as the types of tests to be run in the simulation, input signals, output points, and output content and format. In a preferred embodiment, the simulation environment files will include the monitor and monitoring methods disclosed in the following copending U.S. Patent Applications:

U.S. Pat. App Ser. No.	**Date Filed**	**Title**
09/406,017	9/24/99	Method and Apparatus For a Monitor that Detects and Reports a Status Event to a Database
09/406,016	9/24/99	Method and Apparatus that Reports Multiple Microprocessor Events with a Single Monitor